```
foods = []
prices = []

def add_items():
    while True:
        food = input("Enter Food (q to stop adding): ")
        if food.lower() == "q":
            break
        else:
            try:
                price = float(input(f"Enter the price of {food}: "))
                foods.append(food)
                prices.append(price)
            except ValueError:
                print("Invalid price. Please enter a number.")

def search_item():
    search = input("Enter item to search: ")
    if search in foods:
        count = foods.count(search)
        print(f"'{search}' found {count} time(s).")
        # show prices of found items
        for i, f in enumerate(foods):
            if f == search:
                print(f" - {f} at ${prices[i]:.2f}")
    else:
        print(f"'{search}' not found.")

def remove_item():
    item = input("Enter item to remove: ")
    if item in foods:
        index = foods.index(item)
        foods.pop(index)
        prices.pop(index)
        print(f"'{item}' found and deleted.")
    else:
        print(f"'{item}' not found – deletion unsuccessful.")

def view_sorted():
    if not foods:
        print("No items in the list.")
        return

    # Sort by food names (ascending order)
```

```
    sorted_items = sorted(zip(foods, prices), key=lambda x: x[0])
    print("\n-- Items Sorted by Name --")
    for food, price in sorted_items:
        print(f"{food} - ${price:.2f}")

while True:
    print("\n[ MENU OPTIONS ]")
    print("1 – Add Items")
    print("2 – Search for an Item")
    print("3 – Remove an Item")
    print("4 – View all items (Sorted)")
    print("0 – Exit program")

    choice = input("Pick one [0 to quit]: ")

    if choice == "1":
        add_items()
    elif choice == "2":
        search_item()
    elif choice == "3":
        remove_item()
    elif choice == "4":
        view_sorted()
    elif choice == "0":
        print("Exiting program... Goodbye!")
        break
    else:
        print("Invalid option, please try again.")
```